



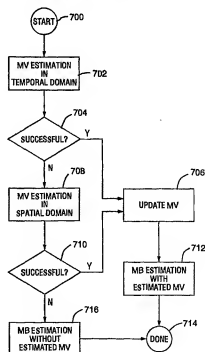
INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification ⁶: H04N 7/36, 5/14	A1	(11) International Publication Number: WO 98/41028 (43) International Publication Date: 17 September 1998 (17.09.98)
(21) International Application Number: PCT/US98/04497 (22) International Filing Date: 6 March 1998 (06.03.98) (30) Priority Data: 08/816,867 13 March 1997 (13.03.97) US (71) Applicant (for all designated States except US): SONY ELECTRONICS INC. [US/US]; 1 Sony Drive, Park Ridge, NJ 07656 (US). (72) Inventors; and (75) Inventors/Applicants (for US only): OZCELIK, Taner [TR/US]; Sony Electronic Inc., 3300 Zanker Road, San Jose, CA 95134-1901 (US). YU, Gong-san [-/US]; Sony Electronics Inc., 3300 Zanker Road, San Jose, CA 95134-1901 (US). GADRE, Shirish, C. [IN/US]; Sony Electronics Inc., 3300 Zanker Road, San Jose, CA 95134-1901 (US). (74) Agents: SAMMUT, Charles, P. et al.; Limbach & Limbach L.L.P., 2001 Ferry Building, San Francisco, CA 94111-4262 (US).	(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, GW, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG). Published <i>With international search report.</i>	

(54) Title: ERROR CONCEALMENT FOR VIDEO IMAGE**(57) Abstract**

A method and apparatus for concealing errors during decoding of a video bit stream utilize estimates (702), if possible, of a motion vector in the temporal domain. The temporal estimation of a lost block uses an estimated motion vector from a block in a subsequent frame, assuming that the movement remains basically constant between two subsequent frames. If estimation in the temporal domain is not possible, motion vectors are estimated (708) in the spatial domain. A macroblock is then estimated based upon the estimated motion vector (712). If estimation in the temporal domain is not possible, macroblock estimation is made without the use of an estimated motion vector (716).

ERROR CONCEALMENT TOP CHART



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroun			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakhstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

ERROR CONCEALMENT FOR VIDEO IMAGE

Background of the Invention**1. Field of the Invention**

5 The present invention relates generally to video encoding and decoding and, in particular, to methods and apparatus for error concealment in video encoding and decoding.

2. Description of the Related Art

10 Advances in audio and video compression and decompression techniques, together with very large scale integration technology, have enabled the creation of new capabilities and markets. These include the storage of digital audio and video in computers and on small optical discs as well as the transmission of digital audio and video signals from direct broadcast satellites.

15 Such advances were made possible, in part, by international standards which provide compatibility between different approaches to compression and decompression. One such standard is known as "JPEG," for Joint Photographic Expert Group. A later developed standard is known as "MPEG 1." This was the first set of standards
20 agreed to by the Moving Pictures Expert Group. Yet another standard is known as "ITU-T H.261", which is a video compression standard particularly useful for video teleconferencing. Although each standard is designed for a specific application, all of the standards have much in common.

25 MPEG1 was designed for storing and distributing audio and motion video, with emphasis on video quality. Its features include random access, fast forward and reverse playback. MPEG1 serves as the basis for video compact disks and for many video games. The original channel bandwidth and image resolution for MPEG1

were established based upon the recording media then available. The goal of MPEG1 was the reproduction of recorded digital audio and video using a 12 centimeter diameter optical disc with a bit rate of 1.416 Mbps, 1.15 Mbps of which are allocated to video.

5 The compressed bit streams generated under the MPEG1 standard implicitly define the decompression algorithms to be used for such bit streams. The compression algorithms, however, can vary within the specifications of the MPEG 1 standard, thereby allowing the possibility of a proprietary advantage in regard to the
10 generation of compressed bit streams.

 A later developed standard known as "MPEG2" extends the basic concepts of MPEG1 to cover a wider range of applications. Although the primary application of the MPEG2 standards is the all digital transmission of broadcast-quality video at bit rates of 4 Mbps
15 to 9 Mbps, it appears that the MPEG2 standard may also be useful for other applications, such as the storage of full length motion pictures on Digital Video Disk ("DVD") optical discs, with resolution at least as good as that presently provided by 12 inch diameter laser discs.

20 The MPEG2 standard relies upon three types of coded pictures. I ("intra") pictures are fields or frames coded as a stand-alone still image. Such I pictures allow random access points within a video stream. As such, I pictures should occur about two times per second. I pictures should also be used where scene cuts (such
25 as in a motion picture) occur.

 P ("predicted") pictures are fields or frames coded relative to the nearest previous I or P picture, resulting in forward prediction processing. P pictures allow more compression than I pictures through the use of motion compensation, and also serve as a
30 reference for B pictures and future P pictures.

B ("bidirectional") pictures are fields or frames that use the most closest (with respect to display order) past and future I or P picture as a reference, resulting in bidirectional prediction. B pictures provide the most compression and increase signal to noise ratio by averaging two pictures.

Such I, P and B pictures are more thoroughly described in U.S. Patent Nos. 5,386,234 and 5,481,553 assigned to Sony Corporation and said U.S. Patents are incorporated herein by reference.

A group of pictures ("GOP") is a series of one or more coded pictures which assist in random accessing and editing. A GOP value is configurable during the encoding process. Since the I pictures are closer together, the smaller the GOP value, the better the response to movement. The level of compression is, however, lower.

In a coded bitstream, a GOP must start with an I picture and may be followed by any number of I, P or B pictures in any order. In display order, a GOP must start with an I or B picture and end with an I or P picture. Thus, the smallest GOP size is a single I picture, with the largest size being unlimited.

In further detail, Figure 1 illustrates a simplified block diagram of an MPEG2 encoder 100. A video stream consisting of macroblock information and motion compensation information is provided to both a discrete cosine transformer 102 and a motion vector generator 104. Each 8 x 8 block (of pixels or error terms) is processed by the discrete cosine transformer 102 to generate an 8 x 8 block of horizontal and vertical frequency coefficients. The quantizer 106 quantizes the 8 x 8 block of frequency-domain error coefficients, thereby limiting the number of allowed values.

Higher frequencies are usually quantized more coarsely than low frequencies, taking advantage of the human perception of quantization error. This results in many frequency-domain error coefficients being zero, especially at higher frequencies.

The output of quantizer 106 is processed by a zigzag scanner 108, which, starting with DC components, generates a linear stream of quantized frequency coefficients arranged in order of increasing frequency. This produces long runs of consecutive zero coefficients, which are sent to the variable length encoder 110.

The linear stream of quantized frequency-domain error coefficients is first run-length encoded by the variable length encoder 110. In the run-length encoding process, the linear stream of quantized frequency-domain error coefficients is converted into a series of run-amplitude (or run-level) pairs. Each pair indicates the number of zero coefficients and the amplitude of the non-zero coefficient which ends the run.

For example, assume a string of error coefficients as follows:

(1) Original error coefficients: 000060000038

Therefore, when this string of error coefficients is variable length encoded, according to the encoding rules described above, the following encoded run-level pairs are obtained:

(2) Encoded run-level pairs: (4,6) (5,3) (0,8)

Of course, as the number of zero coefficients is increased, the error coefficient data will be more effectively compressed by this variable length encoding.

After the variable length encoder 110 encodes the run-level pairs, it then Huffman encodes the run-level pairs. In the Huffman encoding, the run-level pairs are coded differently depending upon whether the run-level pair is included in a list of commonly-occurring run-level pairs. If the run-level pair being Huffman encoded is on the list of commonly-occurring pairs, then it will be encoded into a predetermined variable length code word which corresponds to the run-level pair. If, on the other hand, the run-level pair is not on the list, then the run-level pair is encoded as a predetermined symbol

(such as an escape symbol) followed by a fixed length codes to avoid long code words and to reduce the cost of implementation.

The run-length encoded and Huffman encoded output of the variable length encoder 110 provides a coded video bitstream.

5 Picture type determination circuit 112 determines whether the frame being encoded is a P picture, an I picture or a B picture. In the case of a P or I picture, picture type determination circuit 110 causes the motion vector generator 104 to generate an appropriate motion vector which is then provided to variable length encoder 110. Such
10 motion vector is then coded and combined with the output of variable length encoder 110.

Referring now to Figures 2 and 3, the concept of motion compensation is explained. Motion compensation improves compression of P and B pictures by removing temporal redundancies
15 between pictures. With MPEG 2, it operates at the macroblock level. For example, a previous frame 200 contains, among other macroblocks, a macroblock 202 consisting of 16 pixels (also referred to as "pels") by 16 lines. Motion compensation relies on the fact that, except for scene cuts, most images remain in the same location
20 from frame to frame, whereas others move only a short distance. Thus, such motion can be described as a two-dimensional motion vector that specifies where to retrieve a macroblock from a previously decoded frame to thereby predict the pixel values of a current macroblock. Thus, a macroblock 300 of a current frame 302
25 can be represented by the macroblock 202 (of Figure 2) as modified by a two dimensional motion vector 304. It is to be understood that the macroblock 300 may or may not be within the same boundaries surrounding macroblock 202 in the previous frame 200.

After a macroblock has been compressed using motion
30 compensation, it contains both the prediction (commonly referred to as "motion vectors") and temporal difference (commonly referred to

as "error terms") between the reference macroblock and the macroblock being coded.

Returning to Figure 1, when the coded video bitstream output from variable length encoder 110 is recorded onto a recording medium such as an optical disk, and such recorded information is reproduced for local use, although not completely error free, the decoded (coded) video bit stream is, generally, sufficiently error free so as to not require additional techniques to compensate for errors in the decoded video bit stream. Such a coded video bit stream is typically referred to as a "program stream." When the coded video bitstream output from variable length encoder 110 is transported by, for example, satellite or cable transmission systems, either directly from variable length encoder 110 or from a recording medium onto which the coded video bitstream has been recorded, the probability of errors in the decoded video bitstream increases. Such a coded bitstream is typically referred to as a "transport stream."

Since traditional error detection and correction systems, such as interleaving, require a significant amount of overhead as well as a significant amount of data processing when decoding coded video bitstream signals, current video decoding systems rely upon error *concealment* as opposed to error *correction*. In contrast to error correction, which attempts to reconstruct lost or corrupt data, error concealment aims to generate data which can be substituted for the lost or corrupt data, where any discrepancies in image created by the generated data (generally at the macroblock level) are not likely to be perceived by a viewer of a video image which relies upon such error concealment.

Accordingly, it would be desirable to provide a method and apparatus for concealing errors where the visual effect perceived by a viewer is negligible, and where the method and apparatus adapt to

the different types of available information as may be available to provide such concealment.

Summary of the Invention

5 It is an object of the invention to provide a method and apparatus for concealing errors during decoding of compressed video signals.

It is a further object of the invention to provide a method and apparatus for detecting errors which do not produce illegal syntax.

10 It is a feature of the invention to utilize a temporal prediction of a motion vector to generate a macroblock which will effectively conceal an error in a data stream.

It is a further feature of the invention to compare DC coefficients of a current macroblock to a predicted coefficient to determine whether an error which does not produce illegal syntax
15 has occurred.

It is an advantage of the invention to improve the quality of concealment of an error in a data stream.

It is a further advantage of the invention to improve the quality of detection of an error in a data stream.

20 According to one aspect of the invention, an apparatus for concealing errors includes a detector for detecting the presence of an error in data representing the current macroblock, a system for estimating the at least one motion vector based upon a difference between a forward reference frame at the current macroblock and a
25 decoded motion vector for the forward reference frame at the current macroblock, and a system for estimating the current macroblock based upon the estimated at least one motion vector. According to another aspect of the invention, a method for concealing errors includes the steps of detecting the presence of an
30 error in data representing the current macroblock, estimating the at

least one motion vector based upon a difference between a forward reference frame at the current macroblock and a decoded motion vector for the forward reference frame at the current macroblock, and estimating the current macroblock based upon the estimated at
5 least one motion vector.

These and other objects, features and advantages will become apparent when considered with reference to the following description and the accompanying drawings.

Brief Description of the Drawings

10 Figure 1 is a simplified block diagram of a MPEG 2 video encoder.

Figure 2 is an illustration of a macroblock within a previous frame.

15 Figure 3 is an illustration of a macroblock within a current frame.

Figure 4 is simplified block diagram of a MPEG 2 video decoder of the present invention.

Figure 5 is a block diagram of a motion compensation system of the present invention.

20 Figure 6 is a state diagram which illustrates reference block fetch control of the address generation and control unit of Figure 5.

Figure 7 is a flow chart of a method for estimating macroblocks in accordance with the present invention.

25 Figure 8 is a flow chart of a method for estimating motion vectors in the temporal domain in accordance with the present invention.

Figure 9 is a flow chart of a method for estimating motion vectors in the spatial domain in accordance with the present invention.

Figure 10 is a flow chart of a method for macroblock estimation utilizing estimated motion vectors.

Figure 11 is a flow chart of a method for macroblock estimation without the use of estimated motion vectors.

5 Detailed Description of the Preferred Embodiments

Referring now to Figure 4, a block diagram of an MPEG 2 decoder 400 is shown. Decoder 400 utilizes two internal busses, a GBUS 402 and an RBUS 404. In the preferred embodiment of the invention, GBUS 402 is a 64 bit bus which is utilized for data
10 transfer between DRAM 406 and specific blocks of decoder 400 which are described below. In the preferred embodiment of the invention, DRAM 406 is a static dynamic random access memory, although other types of memories may be utilized. RBUS 404 is an 8 bit bus used primarily for control of specific blocks through
15 reduced instruction set computing ("RISC") CPU 408. RISC CPU 408, which is coupled to both GBUS 402 and RBUS 404, operates to control the functionality of specific blocks, as more particularly described below, as well as performing a portion of video bitstream decoding.

20 Decoder 400 includes a demultiplexer 410 which is coupled to both GBUS 402 and RBUS 404. Similarly, a video decoder 412, an audio decoder 414, a host interface 416, a letter box unit 418, and a sub picture/vertical blanking interval decoder 420 are each coupled to both GBUS 402 and RBUS 404. An RBUS controller 422, an
25 NTSC/PAL encoder 424, a video post filter/on screen display system 426 and an audio clock generator 428 are each coupled to RBUS 404. Audio clock generator 428 outputs a clock signal ACLK. A memory controller 430 is coupled to GBUS 402.

A clock generator 432, which provides a clock signal SCLK, is coupled to host interface 416. An output of letter box unit 418 is provided to video post filter on screen display system 426.

Sub picture/vertical blanking interval decoder 420 is coupled to video post filter on screen display system 426, which system provides its output to NTSC/PAL encoder 424. Sub picture/vertical blanking interval decoder 420 is coupled to video post filter on screen display system 426. A host processor 434 interfaces with host interface 416.

In the preferred embodiment of the invention, when utilized for DVD applications, sub picture/vertical blanking interval decoder 420 and letter box unit 418 are hardwired units. Letter box unit 418 performs a 4-tap vertical filtering and sub-sampling of a video bit stream provided through GBUS 402 and operates to control the video post filter/on screen display system 426. Sub picture/vertical blanking interval decoder 420 operates to decode sub picture ("SP") and vertical blanking interval ("VBI") information in the video bit stream. Typically, a sub picture bitstream consists of subtitles or menu items. For example, this would include karaoke and menu highlighting. Since the VBI bitstream and the SP bitstream are very similar in syntax and functionality (under MPEG 2 specifications), the functionality for decoding both types of bitstreams is incorporated into a single sub picture/vertical blanking interval decoder 420. Thus, in the preferred embodiment of the invention, decoding of the VBI bit stream occurs during the vertical blanking period, while SP bitstream decoding occurs during active display periods.

In the preferred embodiment of the invention, in non-DVD operation, the sub picture/vertical blanking interval decoder 420 decodes and displays on screen display ("OSD") bitstreams. In DVD

operation, however, OSD bitstreams are instead decoded by video post filter on screen display system 426.

RISC CPU 408 operates to parse the video bitstream in order to control the decoder 400. RISC CPU 408 also partially decodes the video bitstream (for example, decoding of top-level data such as headers) and also controls various of the other units within decoder 400 through RBUS 404. A portion of the parsing is also performed by sub picture/video blanking interval decoder 420.

In further detail, RISC CPU 408 can be utilized to change the position of an SP window through RBUS 404. Thus, a user can move the SP window up or down through a command to CPU 404 with a Y coordinate as a parameter.

Letter box unit 418 is essentially a vertical decimation filter with downloadable coefficients. Letter box unit 418 operates to decimate an active area of a frame which has a ratio of 4:3. Therefore, for PAL sequences, letter box unit 418 converts a 720 x 576 frame to a 720 x 432 frame. For NTSC sequences, letter box unit 418 converts a 720 x 480 frame to a 720 x 360 frame. However, in both cases, the active picture area is centered with respect to a display area.

Host processor 434 and RISC CPU 408 utilize DRAM 406 to exchange messages, commands and status information. In the preferred embodiment of the invention, processor 434 and CPU 408 have the capability to interrupt each other. In operation, CPU 408 provides a host command parser to execute such commands from host processor 434. A typical sequence of events during execution of a command by host processor 434 is:

1. Host processor 434 writes a command to DRAM 406 and interrupts CPU 408.

2. CPU 408 reads the command and parameters from DRAM 406.
3. CPU 408 acknowledges the command by writing a status variable to DRAM 406.
- 5 4. Command parser of CPU 408 parses the command and executes it.
5. Optionally, CPU 408 interrupts host processor 434 upon completion of the command to report status.

10 Alternatively, CPU 408 polls a DRAM command buffer (not shown) for every field sync. This buffer is a ring buffer where a write pointer is maintained by host processor 434 while a read pointer is maintained by CPU 408.

Video decoder 412 contains an inverse cosine discrete transformer, a variable length decoder 436, a motion compensation unit 438 and an inverse discrete cosine transformer 440.

15 Video decoder 412 decodes a coded video data stream received through GBUS 402 and provides a decoded stream to NTSC/PAL encoder 424 through RBUS 404. NTSC/PAL encoder converts the decoded stream into an analog signal suitable for display on a television monitor having NTSC and/or PAL signal inputs.

20 Demultiplexer 410 operates on data entering decoder 400. In particular, such data is in the form of packets, and includes audio, video and other streams of multiplexed packets. Demultiplexer 410 selects desired audio packets, video packets and other desired information packets, but rejects the other packets within the video bitstream. For example, audio packets representing audio in several different languages may be present in the video bitstream. Based upon commands from host processor 434, demultiplexer 410 selects only those audio packets corresponding to that language which is
30 selected for presentation with the corresponding video packets.

Host interface 416 provides a glueless interface for host processor 434. RBUS controller 422 sends out messages on RBUS 404 and acts as an arbitrator for RBUS 404. Clock generator 432 provides a clock signal SCLK to the various units within decoder 400, while audio clock generator 428 provides a clock signal ACLK to a digital-to-analog convertor (not shown) which through GBUS 402 receives digital audio signals from audio decoder 414. These digital audio signals are converted to analog signals which can be perceived by humans.

With reference now to Figure 5, the motion compensation system of the video decoder 412 of Figure 4 is now explained. The motion compensation unit 500 includes an address generation and control unit 502. The address generation and control unit 502 corresponds to the memory controller 430 of Figure 4. The address generation and control unit 502 accepts motion vectors from variable length decoder 436 and calculates a starting address of a reference macroblock. Based upon such calculation, the address generation and control unit 502 issues a data transfer request to the memory controller unit 430. In the preferred embodiment of the invention, data transfer occurs in 64 bit (8 byte) segments at addresses aligned at 8-byte boundaries. When this data returns from the DRAM 406, the data are latched within the motion compensation unit 500. Each 8 bit element of these latched data is then run through horizontal and vertical half-pel filters 504, and the resulting data is stored in the prediction RAM (random access memory) 506.

Since the starting address of the reference block may not be aligned at the 8-byte aligned address, multiplexing is required at the input of the prediction RAM 506. For I-pictures, prediction is not required and, therefore, the motion compensation unit sits idle.

For both P and B-pictures, however, prediction data is required for reconstruction by the reconstruction unit 508 of decoded picture data. In the case of B-pictures, the predicted data is obtained by averaging two such predictions, that is, the output of the half-pel
5 filters at the time and a value from a prediction RAM 506 that was stored after a forward prediction. The reconstruction unit 508 supports this averaging of the half-pel filters 504.

An estimation RAM 510 holds coefficient data transformed in the inverse discrete cosine transformer 440. Reconstruction of each
10 picture starts once the estimation RAM 510 is full. The motion compensation unit 500, at such a time, issues a data transfer request and begins reconstruction. The reconstruction basically consists of adding signed numbers from the output of the inverse discrete cosine transformer stored in the estimation RAM 510 to the
15 outputs (stored in the prediction RAM 506) of the half-pel filters 504 for non-intra blocks. For intra-blocks however, the addition is not required. In such a case, the adder output is clipped before it is latched at the output of the reconstruction unit 508 when reconstruction of the picture occurs.

Referring now to the state diagram of Figure 6, a state
20 machine 600 which represents the functionality of the address generation and control unit 502 in regard to the transfer of reference picture data from DRAM 406 and the construction of a macroblock is now explained. From a start state 602, a state machine 600
25 proceeds to a get address state 604. If no motion compensation is to be used to construct the macroblock, state machine 600 proceeds to a y0_wait state 606. If only backward motion compensation is to be utilized, then state machine 600 proceeds to state 608 to get or
30 fetch a previous macroblock b which will serve as a reference macroblock. If however forward motion compensation is to be utilized, then state machine 600 proceeds to state 610 to get or

fetch a forward macroblock f which will serve as a reference macroblock. State machine 600 then proceeds to the y0_wait state 606. If the macroblock to be constructed is to be based upon both the forward macroblock f and the previous macroblock b, then state machine 600 proceeds from state 610 to state 608 to also get or fetch a previous macroblock. In such an instance, both the forward and the previous macroblock will serve as reference macroblocks.

At state 606, state machine 600 waits for luminance data to be received in regard to the reference macroblock or macroblocks. At state 612 the luminance portion of the macroblock to be constructed is reconstructed. At state 614 state machine waits for chrominance data to be received in regard to the reference macroblock or macroblocks. At state 618 reconstruction of the chrominance portion of the macroblock to be constructed occurs. Upon completion of chrominance data reconstruction, state machine 600 proceeds to state 620 to await an instruction to construct a new macroblock.

Similar to the case of the previously constructed macroblock, state machine 600 then proceeds to a get address 1 state 622. if no motion compensation is to be used to construct the macroblock, state machine 600 proceeds to a y1_wait state 624. If only backward motion compensation is to be utilized, then state machine 600 proceeds to state 626 to get or fetch a previous macroblock b1 which will serve as a reference macroblock. If, however, forward motion compensation is to be utilized, then state machine 600 proceeds to state 628 to get or fetch a forward macroblock f1 which will serve as a reference macroblock. State machine 600 then proceeds to the y1_wait state 624. If the new macroblock to be constructed is to be based upon both the forward macroblock f1 and the previous macroblock b1, state machine 600 proceeds from state 628 to state 626 to also get or fetch the previous macroblock b1.

In such an instance both the forward macroblock f1 and the previous macroblock b1 will serve as reference macroblocks.

At state 624, state machine 600 waits for luminance data to be received in regard to the reference macroblock or macroblocks.

5 At state 630 the luminance portion of the macroblock to be constructed is reconstructed. At state 632 state machine waits for chrominance data to be received in regard to the reference macroblock or macroblocks. At state 634 reconstruction of the chrominance portion of the macroblock to be constructed occurs.

10 Upon completion of such chrominance data reconstruction, state machine 600 proceeds to back to start state 602.

As the state transitions of Figure 6 depict, once an address is obtained as in state 604 or 622, sampling occurs to determine whether or not motion compensation is required for a macroblock.

15 For coded pictures that require motion compensated reference calculations, state machine 600 waits until the motion vector FIFO memory of the variable length decoder 436 is not empty. The address generation and control unit 502 then generates a request for a motion vector. Two consecutive requests, one for X (horizontal)

20 and one for Y (vertical) components of the motion vectors are made. Once the address generation and control unit 502 obtains both components of the motion vector, the address of the reference block is calculated. The address generation and control unit 502 then sends a request for data transfer to the memory controller unit.

25 As mentioned above, when a motion vector points to a sub-pixel location instead of to an exact pixel location, in order to more accurately represent a P or B picture, it is necessary to generate half-pixel (half-pel) data.

In the event that during transport (or even during local reproduction) of a recorded coded video bit stream, errors are

30 detected in the video bit stream in regard to a particular macroblock.

In the preferred embodiment of the invention, the smallest unit of concealment is a slice. A slice consists of a series of sequential macroblocks. In order to provide such concealment, motion vectors are estimated using either temporal prediction or spatial prediction.

5 In the case of spatial prediction, pixels from a successfully decoded macroblock are copied for use in decoding the macroblock having a data error. In the case of temporal prediction, motion vectors from a successfully decoded macroblock are utilized to predict a new motion vector field in order to decode the macroblock having a data error.

10 In further detail, if in decoding a coded video bitstream, a frame K is missing a macroblock or a portion of a macroblock (as could result from a data error), the basic concept is that if there is a motion of an object from a frame K-2 (that is, two frames prior to frame K), one can assume that this motion will most likely continue from frame K-2 up through frame K. Therefore, the assumption is that the motion will be basically linear. Based upon that assumption, the present invention estimates pixels and motion vectors, the estimation method depending upon the data available for such estimation.

20 In utilizing the estimated pixels and/or motion vectors, actual concealment is delayed until the occurrence of a later slice. When an error is detected, such error is recorded together with its location. In particular, data regarding the error and its location is written into a register and, after a second or third subsequent slice, an interrupt command is issued to allow processing of the macroblock utilizing estimated pixels and/or motion vectors. In the preferred embodiment of the invention, when applied to interlaced video, although four motion vectors per macroblock are available for use, only two motion vectors are utilized, as will be detailed below.

Referring now to Figure 7, the error concealment of the present invention is explained in further detail. Once an error has been detected and an interrupt command has been issued, the error concealment algorithm starts at step 700. The motion compensation unit 438 first attempts to estimate motion vectors in the temporal domain at step 702. Figure 8 illustrates such a method. The algorithm starts at step 800. At step 802, the motion compensation unit 438 determines whether a decoded motion vector for a forward reference frame at the macroblock positioned by a vector p is available. This motion vector is designated as $MV(k-m, \vec{p})$, where k is the current frame, and m is the frame index difference between the current frame and a forward reference frame. If not, motion vector estimation in the temporal domain is not performed and the algorithm proceeds to step 804, which indicates a failed attempt. If a decoded motion vector for a forward reference frame at the macroblock positioned by the vector p is available, the algorithm proceeds to step 806, which determines whether a decoded motion vector is available for the difference between (1) a forward reference frame at the macroblock positioned by the vector p ; and (2) a decoded motion vector for a forward reference frame at the macroblock positioned by the vector p , where such decoded motion vector is designated by $MV(k-m, \vec{p})$. If not available, the algorithm proceeds to step 804 to indicate a failed attempt. If available, the algorithm proceeds to step 808 at which an estimated motion vector for a current frame, the k -th frame, at the macroblock positioned by the vector p is determined. Such estimated motion vector is taken to be equal to the difference between (1) a forward reference frame at the macroblock positioned by the vector p ; and (2) a decoded motion vector for a forward reference frame at the macroblock positioned by the vector p .

The algorithm then proceeds to step 810 which indicates a successful motion vector estimation in the temporal domain.

Returning to Figure 7, at step 704, it is determined whether motion vector estimation in the temporal domain was successful.

5 If so, the algorithm proceeds to step 706, where based upon the estimated motion vector, the motion vector to be used for estimating the subject macroblock is updated. If the motion vector estimation in the temporal domain was not successful, the algorithm proceeds to step 708, where motion vector estimation is performed in the spatial domain. The algorithm for such estimation is shown in Figure 9. The algorithm starts at step 900 of Figure 9 and proceeds to step 902, where it is determined whether a decoded motion vector for the macroblock located immediately above the estimating macroblock is available. Such a motion vector is designated by
10 $MV(k, \vec{p}-(1,0))$. If not, a failure is indicated at step 904. If a decoded motion vector for the macroblock located immediately above the estimating macroblock is available, then, at step 906, the motion vector for the current frame, the k-th frame at the macroblock positioned by the vector \vec{p} , $\sim MV(k, \vec{p})$, is estimated to be equal to the decoded motion vector, the macroblock located
20 immediately above the estimating macroblock, $MV(k, \vec{p}-(1,0))$, where $(1,0)$ is a vector indicating a row index as 1 and a column index as 0. The algorithm then proceeds to step 908 which indicates a successive motion vector estimation in the spatial domain.

25 Returning again to Figure 7, at step 710, a determination is made whether or not a motion vector estimation has been successfully made in the spatial domain. If such a determination has been made (step 908), then the motion vector for the current macroblock is updated at step 706. Then, at step 712, the current
30 macroblock is estimated using the just estimated motion vector,

whether that motion vector is estimated in the temporal domain at step 702 or, in the spatial domain, at step 708.

As shown in Figure 10, at step 1000, macroblock estimation with the estimated motion vector is started. At step 1002, the
5 estimated macroblock for the current frame, the k-th frame, at the macroblock positioned by the estimated motion vector \vec{p} , $\sim MB(k, \vec{p})$, is estimated to be equal to the decoded macroblock of the difference of (1) the forward reference frame at the macroblock positioned by the vector \vec{p} ; and (2) the estimated motion vector for the current
10 frame, the k-th frame, at the macroblock positioned by the vector \vec{p} . This decoded macroblock is designated as $MB(k-m, \vec{p} - \sim MV(k, \vec{p}))$, where m is the frame index difference between the current frame and a forward reference frame. Upon completion of estimation of the current macroblock at step 712, the algorithm is completed at
15 step 714.

Returning to step 710 of Figure 7, if motion vector estimation in the spatial domain is not successful, then, at step 716, the current macroblock is estimated without the use of an estimated motion vector. Step 716 is detailed in Figure 11. In Figure 11,
20 macroblock estimation without use of an estimated motion vector starts at step 1100. At step 1102, it is determined whether the macroblock for the frame preceding the current frame (the k-th frame being the current frame) positioned by the vector \vec{p} , $MB(k-1, \vec{p})$ is available. If such macroblock is available, then, at step 1104, the
25 current macroblock positioned by the vector \vec{p} is estimated to be equal to the macroblock for the frame preceding the current frame positioned by the vector \vec{p} . The algorithm is then completed as indicated at step 714.

If the macroblock for the frame preceding the current frame
30 (the k-th frame being the current frame) positioned by the vector \vec{p} is not available, then, at step 1106, it is determined whether the

macroblock for the current frame positioned by the vector p but indexed by minus 1 row and in the same column, $MB(k, \vec{p} - (1, 0))$ is available, where $(1, 0)$ is a vector indicating a row index as 1 and a column index as 0. If such a macroblock is available, then, at step 1108, the current macroblock (for the current frame, the k -th frame, positioned by the vector p) is estimated to be equal to the macroblock for the current frame positioned by the vector p but indexed by minus 1 row and in the same column, $MP(k, \vec{p} - (1, 0))$. The algorithm is then completed as indicated at step 714.

10 If it is determined that the macroblock for the current frame positioned by the vector p but indexed by minus 1 row and in the same column, is not available, then, at step 1110, it is then determined whether the decoded macroblock for the macroblock located immediately above the macroblock to be estimated, 15 $MB(k, p + (1, 0))$, is available, where $(1, 0)$ is a vector indicating a row index as 1 and a column index as 0. If the decoded macroblock for the macroblock located immediately above the macroblock to be estimated is available, then, at step 1112, the estimated macroblock for the current frame, the k -th frame, at the macroblock positioned 20 by the vector p is estimated to be equal to such decoded macroblock for the macroblock located immediately above the macroblock to be estimated, $MB(k, \vec{p} + (1, 0))$. The algorithm is then completed as indicated at step 714. In the event that the decoded macroblock for the macroblock located immediately above the macroblock to be 25 estimated is not available, then the macroblock estimation without an estimated motion vector fails, as indicated at step 1114. In this case, the macroblock can be left blank.

30 It is to be understood that, although the present invention has been described in relation to decoding of a coded video bit stream, the present invention is also applicable to the coding of a video bit

stream, where an error is detected during or after coding and the error is concealed prior to recording or transport.

5 Although only certain embodiments have been described in detail, those having ordinary skill in the art will certainly understand that many modifications are possible without departing from the teachings hereof. All such modifications are intended to be encompassed within the following claims.

WE CLAIM:

1. A method of concealing errors while decoding a coded video bit stream, where a current macroblock is represented by data of reference frame and at least one motion vector, comprising the steps of:

detecting (700) the presence of an error in data representing the current macroblock;

estimating the at least one motion vector based upon a difference between a forward reference frame at the current macroblock and a decoded motion vector for the forward reference frame at the current macroblock (702); and

estimating the current macroblock based upon the estimated at least one motion vector (712).

2. The method of claim 1, wherein the estimated at least one motion vector is equal to the difference between a forward reference frame at the current macroblock and a decoded motion vector for the forward reference frame at the current macroblock (808).

3. The method of claim 1, wherein the step of estimating the current macroblock based upon the estimated at least one motion vector comprises the step of:

estimating the current macroblock at a macroblock positioned by the estimated motion vector to be equal to the decoded macroblock of the difference of the forward reference frame at the macroblock positioned by the vector and the estimated motion vector for the current frame at the macroblock positioned by the vector (1002).

4. A method of concealing errors while decoding a coded video bit stream, where a current macroblock is represented by data of reference frame and at least one motion vector, comprising the steps of:

- 5 detecting the presence of an error in data representing the current macroblock (700);
- estimating the at least one motion vector based upon a decoded motion vector for a macroblock located immediately above the current macroblock (708); and
- 10 estimating the current macroblock based upon the estimated at least one motion vector (712).

5. The method of claim 4, wherein the estimated at least one motion vector is equal to the decoded motion vector for a macroblock located immediately above the current macroblock (906).

15

6. The method of claim 4, wherein the step of estimating the current macroblock based upon the estimated at least one motion vector comprises the step of:

- estimating the current macroblock at a macroblock
- 20 positioned by the estimated motion vector to be equal to the decoded macroblock of the difference of the forward reference frame at the macroblock positioned by the vector and the estimated motion vector for the current frame at the macroblock positioned by the vector (1002).

25

7. A method of concealing errors while decoding a coded video bit stream, where a current macroblock is represented by data of reference frame and at least one motion vector, comprising the steps of:

detecting the presence of an error in data representing the current macroblock (1100); and

5 if a macroblock for a frame preceding a current frame positioned by a vector is available (1002), estimating the current macroblock positioned by the vector to be equal to the macroblock for the frame preceding the current frame positioned by the vector (1004); otherwise (1002)

10 if a macroblock for the current frame positioned by the vector p but indexed by minus 1 row and in the same column is available (1106), estimating the current macroblock to be equal to the macroblock for the current frame positioned by the vector p but indexed by minus 1 row and in the same column (1108); otherwise (1106),

15 if a decoded macroblock for the macroblock located immediately above the current macroblock is available (1110), estimating the current macroblock for the current frame to be equal to such decoded macroblock for the macroblock located immediately above the macroblock to be estimated (1112).

20 8. A method of concealing errors while decoding a coded video bit stream, where a current macroblock is represented by data of reference frame and at least one motion vector, comprising the steps of:

25 detecting the presence of an error in data representing the current macroblock (700);

30 if a difference between a forward reference frame at the current macroblock and a decoded motion vector for the forward reference frame at the current macroblock is available (806), estimating the at least one motion vector to be equal to a difference between a forward reference frame at the current macroblock and a decoded motion vector for the forward

reference frame at the current macroblock (808); and, if not (806),

5 if a decoded motion vector for a macroblock located immediately above the current macroblock is available (902), estimating the at least one motion vector to be equal to decoded motion vector for a macroblock located immediately above the current macroblock (906);

10 estimating the current macroblock at a macroblock positioned by the estimated motion vector to be equal to the decoded macroblock of the difference of the forward reference frame at the macroblock positioned by the vector and the estimated motion vector for the current frame at the macroblock positioned by the vector (1002); and if a decoded motion vector for a macroblock located immediately above the current macroblock is not available (710),

15 if a macroblock for a frame preceding a current frame positioned by a vector is available (1102), estimating the current macroblock positioned by the vector to be equal to the macroblock for the frame preceding the current frame positioned by the vector (1104); and, if not (1102),

20 if a macroblock for the current frame positioned by the vector p but indexed by minus 1 row and in the same column is available (1106), estimating the current macroblock to be equal to the macroblock for the current frame positioned by the vector p but indexed by minus 1 row and in the same column (1108); and, if not (1106),

25 if a decoded macroblock for the macroblock located immediately above the current macroblock is available (1110), estimating the current macroblock for the current frame to be equal to such decoded macroblock for the macroblock located immediately above the macroblock to be estimated (1112).

30

9. An apparatus for concealing errors while decoding a coded video bit stream, where a current macroblock is represented by data of reference frame and at least one motion vector, comprising:

- 5 a detector for detecting the presence of an error in data representing the current macroblock (700);
 means for estimating the at least one motion vector based upon a difference between a forward reference frame at the current macroblock and a decoded motion vector for the
10 forward reference frame at the current macroblock (702); and
 means for estimating the current macroblock based upon the estimated at least one motion vector (712).

10. The apparatus of claim 9, wherein the estimated at least one motion vector is equal to the difference between a forward
15 reference frame at the current macroblock and a decoded motion vector for the forward reference frame at the current macroblock (808).

11. The apparatus of claim 9, wherein the means for estimating the current macroblock based upon the estimated at least
20 one motion vector comprises:

- means for estimating the current macroblock at a macroblock positioned by the estimated motion vector to be equal to the decoded macroblock of the difference of the
 forward reference frame at the macroblock positioned by a
25 vector and the estimated motion vector for the current frame at the macroblock positioned by the vector (1002).

12. An apparatus for concealing errors while decoding a coded video bit stream, where a current macroblock is represented

by data of reference frame and at least one motion vector,
comprising:

a detector for detecting the presence of an error in data
representing the current macroblock (700);

- 5 means for estimating the at least one motion vector
based upon a decoded motion vector for a macroblock located
immediately above the current macroblock (708); and
means for estimating the current macroblock based
upon the estimated at least one motion vector (712).

- 10 13. The apparatus of claim 12, wherein the estimated at
least one motion vector is equal to the decoded motion vector for a
macroblock located immediately above the current macroblock
(906).

- 15 14. The apparatus of claim 12, wherein the means for
estimating the current macroblock based upon the estimated at least
one motion vector comprises:

- means for estimating the current macroblock at a
macroblock positioned by the estimated motion vector to be
equal to the decoded macroblock of the difference of the
20 forward reference frame at the macroblock positioned by a
vector and the estimated motion vector for the current frame
at the macroblock positioned by the vector (1002).

- 25 15. An apparatus for concealing errors while decoding a
coded video bit stream, where a current macroblock is represented
by data of reference frame and at least one motion vector,
comprising:

a detector for detecting the presence of an error in data
representing the current macroblock (1100);

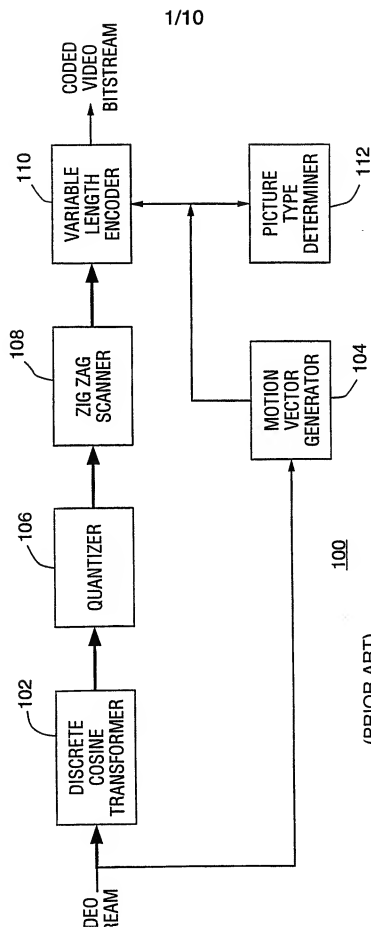
means for determining if a macroblock for a frame preceding a current frame positioned by a vector is available (1102) and, if available, estimating the current macroblock positioned by the vector to be equal to the macroblock for the frame preceding the current frame positioned by the vector (1104), otherwise (1102), if a macroblock for the current frame positioned by the vector p but indexed by minus 1 row and in the same column is available (1106), estimating the current macroblock to be equal to the macroblock for the current frame positioned by the vector p but indexed by minus 1 row and in the same column (1108), otherwise (1106), if a decoded macroblock for the macroblock located immediately above the current macroblock is available (1110), estimating the current macroblock for the current frame to be equal to such decoded macroblock for the macroblock located immediately above the macroblock to be estimated (1112).

16. An apparatus for concealing errors while decoding a coded video bit stream, where a current macroblock is represented by data of reference frame and at least one motion vector, comprising:

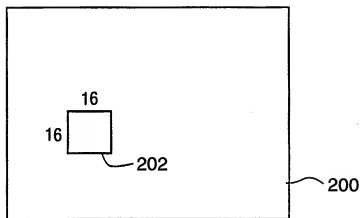
a detector for detecting the presence of an error in data representing the current macroblock (700);
means for determining if a difference between a forward reference frame at the current macroblock and a decoded motion vector for the forward reference frame at the current macroblock is available (806), and estimating the at least one motion vector to be equal to a difference between a forward reference frame at the current macroblock and a decoded motion vector for the forward reference frame at the current macroblock (808); and, if not available (806), if a

decoded motion vector for a macroblock located immediately above the current macroblock is available (902), estimating the at least one motion vector to be equal to decoded motion vector for a macroblock located immediately above the current macroblock (906); and

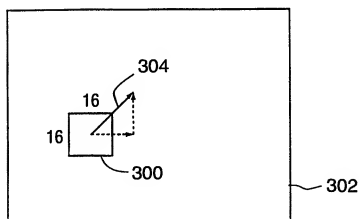
means for estimating the current macroblock at a macroblock positioned by an estimated motion vector to be equal to the decoded macroblock of the difference of the forward reference frame at the macroblock positioned by the vector and the estimated motion vector for the current frame at the macroblock positioned by the vector (1002); and, if a decoded motion vector for a macroblock located immediately above the current macroblock is not available (710), if a macroblock for a frame preceding a current frame positioned by a vector is available (1102), estimating the current macroblock positioned by the vector to be equal to the macroblock for the frame preceding the current frame positioned by the vector (1104); and, if not available (1102), if a macroblock for the current frame positioned by the vector p but indexed by minus 1 row and in the same column is available (1106), estimating the current macroblock to be equal to the macroblock for the current frame positioned by the vector p but indexed by minus 1 row and in the same column (1108); and, if not available (1106), if a decoded macroblock for the macroblock located immediately above the current macroblock is available (1110), estimating the current macroblock for the current frame to be equal to such decoded macroblock for the macroblock located immediately above the macroblock to be estimated (1112).

**FIG. 1**

2/10

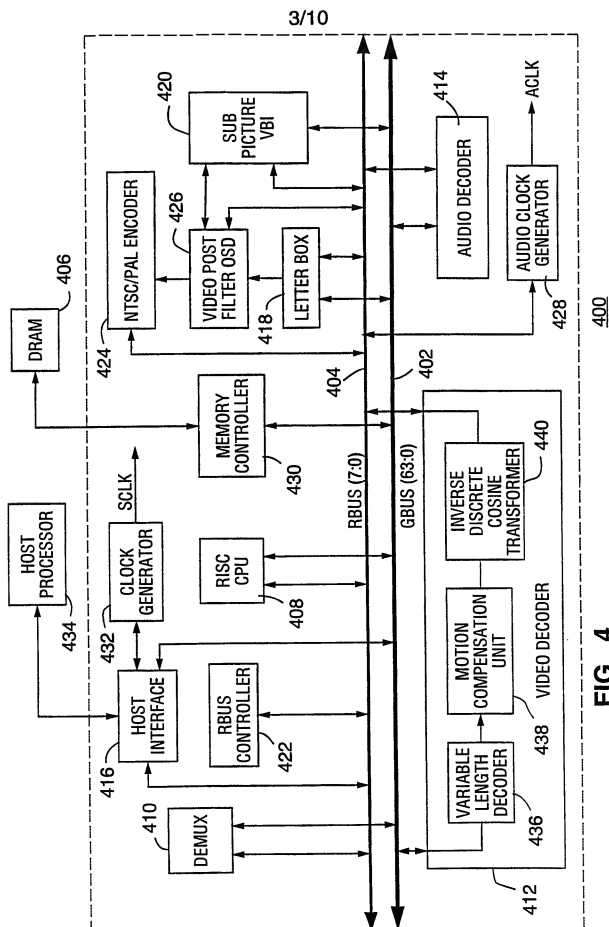


PREVIOUS FRAME

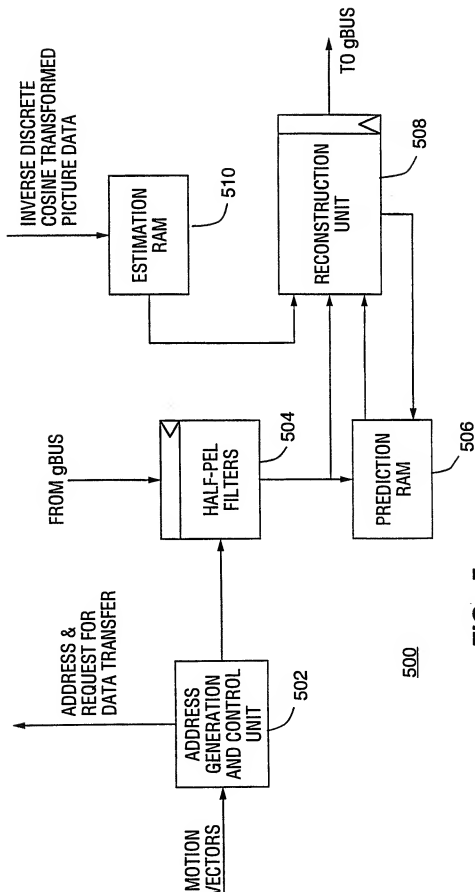
FIG. 2

CURRENT FRAME

FIG. 3



4/10

**FIG. 5**

5/10

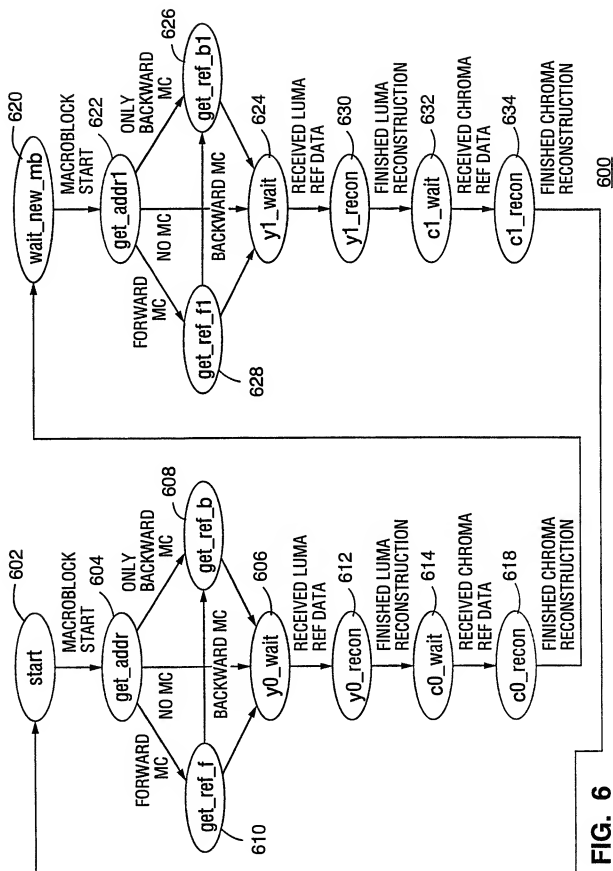


FIG. 6

6/10

ERROR CONCEALMENT TOP CHART

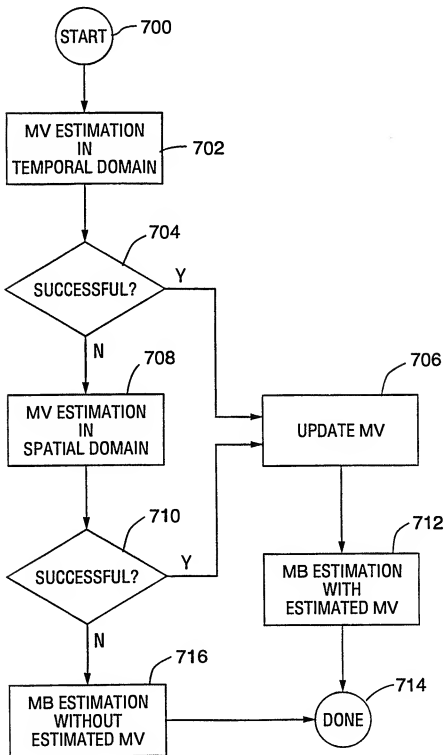
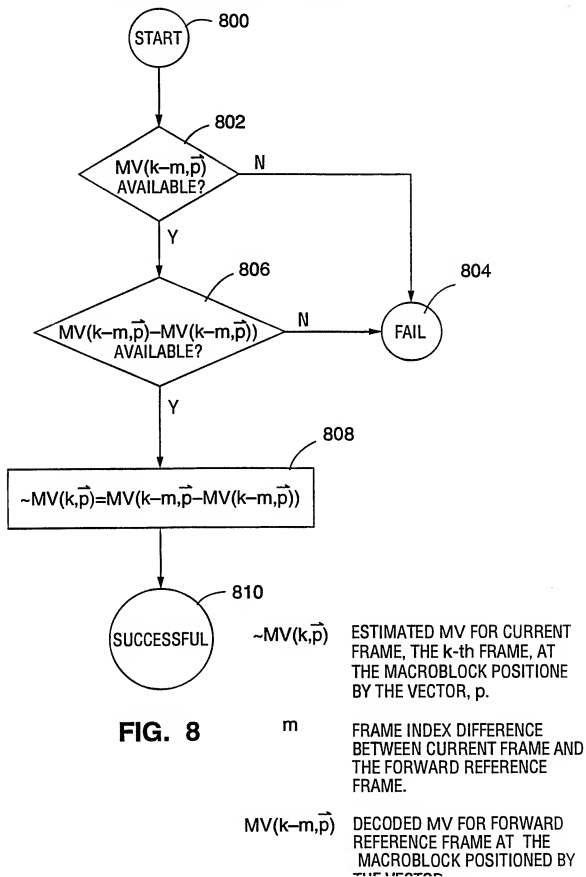


FIG. 7

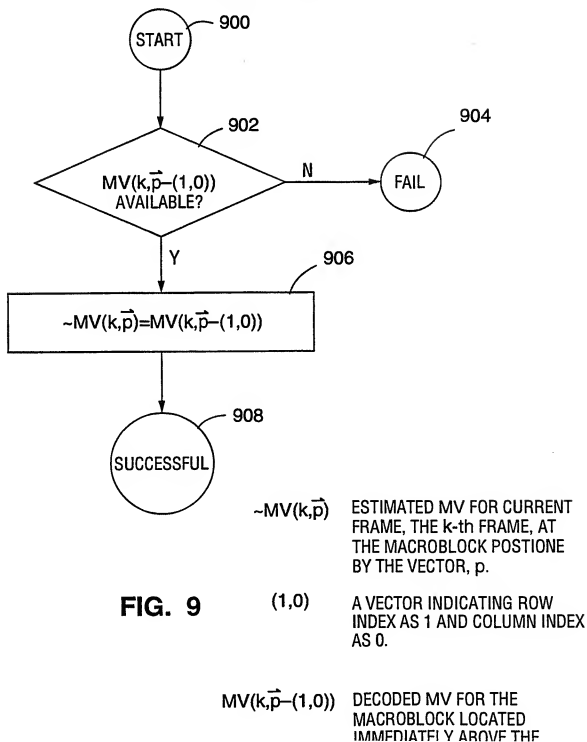
7/10

MV ESTIMATION IN TEMPORAL DOMAIN



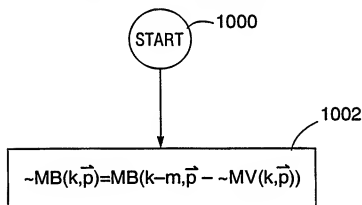
8/10

MV ESTIMATION IN SPATIAL DOMAIN



9/10

MB ESTIMATION WITH ESTIMATED MV



$\sim MV(k, \vec{p})$ ESTIMATED MV FOR CURRENT
FRAME, THE k -th FRAME, AT
THE MACROBLOCK POSITIONED
BY THE VECTOR, p .

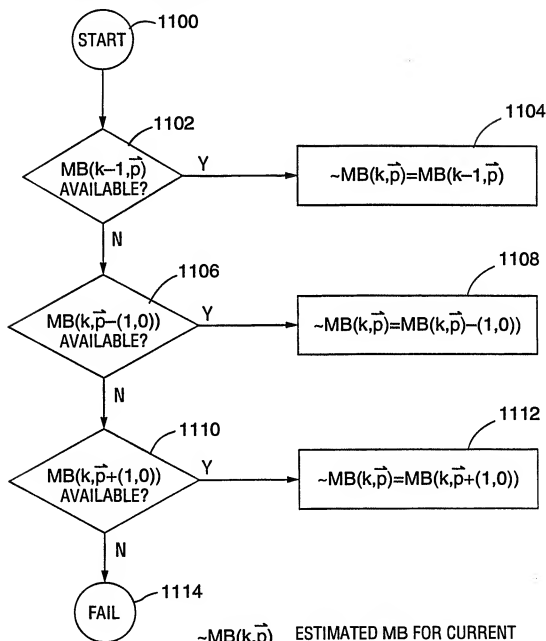
m FRAME INDEX DIFFERENCE
BETWEEN CURRENT FRAME AND
THE FORWARD REFERENCE
FRAME.

$MB(k-m, \vec{p})$ DECODED MB ON FORWARD
REFERENCE FRAME AT THE
MACROBLOCK POSITIONED BY
THE VECTOR, p .

FIG. 10

10/10

MB ESTIMATION IN ESTIMATED MV



ESTIMATED MB FOR CURRENT
FRAME, THE k-th FRAME, AT
THE MACROBLOCK POSTIONE
BY THE VECTOR, \vec{p} .

FIG. 11

(1,0)

A VECTOR INDICATING ROW
INDEX AS 1 AND COLUMN INDEX
AS 0.

 $\text{MB}(k, \vec{p} - (1, 0))$

DECODED MB FOR THE
MACROBLOCK LOCATED
IMMEDIATELY ABOVE THE

INTERNATIONAL SEARCH REPORT

Internat. Application No.
PCT/US 98/04497

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 H04N7/36 H04N5/14

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 247 363 A (SUN HUIFANG ET AL) 21 September 1993 see abstract see column 10, line 19 - line 59; figure 4A ---	1-16
A	EP 0 727 910 A (THOMSON MULTIMEDIA SA) 21 August 1996 see abstract see page 3, line 44 - page 4, line 19; figures 3,6 --- -/--	1-16

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents:

- *A* document defining the general state of the art which is not considered to be of particular relevance
- *E* earlier document but published on or after the international filing date
- *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- *O* document referring to an oral disclosure, use, exhibition or other means
- *P* document published prior to the international filing date but later than the priority date claimed

- *T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- *X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- *Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
- *Z* document member of the same patent family

Date of the actual completion of the international search

24 April 1998

Date of mailing of the international search report

22. 05. 98

Name and mailing address of the ISA
European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Farman, T

INTERNATIONAL SEARCH REPORT

Internat. Application No.
PCT/US 98/04497

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	LEE S H ET AL: "TRANSMISSION ERROR DETECTION, RESYNCHRONIZATION, AND ERROR CONCEALMENT FOR MPEG VIDEO DECODER" PROCEEDINGS OF THE SPIE, vol. 2094, no. PART 01, 8 November 1993, pages 195-204, XP002043758 see page 4TH, last paragraph ---	1-16
A	SUN H ET AL: "ADAPTIVE ERROR CONCEALMENT ALGORITHM FOR MPEG COMPRESSED VIDEO" VISUAL COMMUNICATIONS AND IMAGE PROCESSING 18-20 NOVEMBER 1992, BOSTON, US, vol. 1818, no. PART 02, 18 November 1992, pages 814-824, XP002043757 see page 4TH, paragraph 2 ---	1-16
X	US 5 561 532 A (OHNISHI SHINJI ET AL) 1 October 1996 see column 11, line 64 - column 12, line 20 see column 5, line 16 - line 31; figure 3 see column 6, line 10 - line 22 ---	4,5,12, 13
A	US 5 552 831 A (MACHIDA YUTAKA ET AL) 3 September 1996 see abstract see column 2, line 34 - line 49 see column 4, line 8 - line 58 -----	4,5,12, 13

INTERNATIONAL SEARCH REPORT

Information on patent family members

Internat. Application No

PCT/US 98/04497

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5247363 A	21-09-93	DE 4305578 A JP 6070306 A	09-09-93 11-03-94
EP 0727910 A	21-08-96	US 5621467 A CN 1141561 A JP 8256311 A	15-04-97 29-01-97 01-10-96
US 5561532 A	01-10-96	JP 6292181 A JP 6303597 A JP 7030848 A	18-10-94 28-10-94 31-01-95
US 5552831 A	03-09-96	JP 6020050 A AU 4171693 A CA 2099407 A EP 0577417 A	28-01-94 06-01-94 04-01-94 05-01-94